

Sandbox SK příručka pro službu Iniciování platby

Change log

Date	Version	Description
29.01.2020	1	První verze dokumentu
19.05.2020	2	Úprava dokumentu
18.11.2020	3	Úprava dokumentu (doplněna některá specifika platebního styku na Slovensku, aj.)

Obsah

1.	Služba Inicie platby (PIS)	4
2.	Sandbox API pro volání služby Inicie platby	4
3.	Vydání certifikátu	4
4.	Definice mocku	4
5.	Hlášení chyb	5
6.	Postup pro vygenerování klíče/tokenu ke službě PISP	5
7.	Způsoby volání mocku PISP	8
8.	Postup přístupu do aplikace přes konzoli na API portálu	9
	Volání mocku PIS „New payment“ na otestování	9
	Volání mocku PIS „Payment status information“ k otestování	12
9.	Postup přístupu do aplikace přes přímé volání	20
	Nová platba – I. iniciace platby (POST /my/payments)	20
	Elementy zprávy Nová platba – iniciace platby	21
	Elementy response Nová platba - iniciace platby	32
	Status založené/iniciované platby (GET /payments/{paymentId}/status).....	33
	Elementy zprávy Status založené/iniciované platby	33
	Krok II. Inicie autorizace platby – specifické pro každou banku (POST /my/payments/{paymentId}/sign/{signId}).	35
	Elementy zprávy Krok II. Inicie autorizace platby - specifické pro každou banku	36
	Dotaz na dostatek prostředků (POST /accounts/balanceCheck)	37
	1.1. Elementy zprávy Dotaz na dostatek prostředků	38
	1.2. Elementy response Dotaz na dostatek prostředků	38
	1.3. Návrátové kódy parametru „response“ – Code set:	38
10.	PSD2 slovník - výběr	39

1. Služba Iniciační platby (PIS)

Přehled implementovaných zdrojů:

- iniciační platby –
 - o Krok I. Iniciační autorizované platby
- status založené/iniciované platby – umožňuje získat informace o stavu platby během jejího zpracování v KB
- autorizace platby
 - o Krok II. Iniciační autorizované platby – pomocí tohoto resource je iniciována autorizace platby

Nepovolené typy plateb pro prostředí Sandbox:

- Trvalé příkazy
- Inkasa (povolení k inkasu, příkaz k inkasu)
- Dávky platebních příkazů
- Instantní platby
- Platba šekem

Komerční banka vychází z jednotné struktury a formátu informací, která byla vydefinována Českou bankovní asociací v dokumentu [Czech Open Banking Standard](#).

Informace poskytované prostřednictvím Otevřeného bankovní API jsou lokalizovány do českého i anglického jazyka.

Povolená znaková sada vychází ze pro SWIFT a pro zahraniční/SEPA platby.

Během jednoho volání může být zaslán a zpracován vždy jen jeden dotaz.

2. Sandbox API pro volání služby Iniciační platby

Sandbox umožňuje třetím stranám vyzkoušení služby (mock) pro získávání níže uvedených informací, vztahujících se k platebnímu účtu klienta Komerční banky, a.s., pobočky zahraniční banky (dále jen Komerční banky).

Do sandboxu API mohou přistupovat jakékoliv subjekty, tedy ne jen třetí strany s licencí pro PSD2 služby. Nutnou podmínkou je registrace na API portálu KB <https://api.koba.sk/portal/?tenant=api.kb.sk>. Bez registrace není možné služby na sandboxu využívat. Postup registrace je popsán v dokumentu API Sandbox Registrace.

3. Vydání certifikátu

Certifikát je nezbytnou podmínkou jak pro produkční volání, tak pro sandbox SK PSD2. Pro využití na sandboxu budou třetím stranám po jejich registraci na Sandboxu poskytnuty certifikáty od Komerční banky **na základě žádosti zasláné na emailovou schránku api@kb.cz**. Certifikáty ze sandboxu nejsou určeny pro produkční využití. Na produkci jsou taková volání zamítána a monitorována. Na produkci se používají kvalifikované PSD2 certifikáty vydané kvalifikovanou certifikační autoritou dle EU QTSP seznamu na <https://webgate.ec.europa.eu/tl-browser/>

4. Definice mocku

Jsou definovány vzorové requesty pro statické volání, které není možné ve volání třetí stranou měnit. V případě změny bude request zamítnut. Dle použitých vzorových requestů jsou vráceny odpovídající response, případně error kódy.

1. Iniciační platby:

- Iniciační platby – pro získání vstupních hodnot, tedy validní struktury iniciované platby je nutné kliknout na šedé pole **Example Value** nacházející se vpravo od pole paymentRequest. Sandbox automaticky těmito hodnotami vyplní pole pro zadání platby.
- Iniciační platby platby - error na nevalidní IBAN 400 - AC02 - [InvalidDebtorAccountNumber], pole debtorAccount.identification.iban
- Iniciační platby platby - error na nevalidní částku 400 - AM12 - [InvalidAmount], pole (amount.instructedAmount.value)
- Iniciační platby platby - error na nevalidní datum splatnosti, datum splatnosti je v budoucnosti, 400 NARR - [Entered Execution date cannot exceed actual Business date], pole requestedExecutionDate
- Iniciační platby platby - error na nevalidní formáty polí - vrácení kolekce errorů

2. Status platby:

- Status platby – jako vstupní hodnotu paymentID (transactionID) je nutné vyplnit **ZU000007AX3**
- Status platby - error na neexistenci paymentID (transactionID), pokud není ID nalezeno, je vygenerován error 404 _NOT_FOUND

5. Hlášení chyb

Hlášení chyb Sandboxu nebo jeho volání probíhá vždy pomocí mailové schránky api@kb.cz. Odeslaný mail musí obsahovat níže uvedené náležitosti v případě chybějící požadované informace nebude možné dotaz nebo chybu zpracovat.

PSD2 API: CZ, SK

Prostředí: Sandbox, Produkce

Zda bylo voláno z FE Sandbox vč. typu a verze použitého prohlížeče nebo v případě BE volání název a verzi programu pro BE volání

Typ volání

Datum a čas uskutečněního volání

IP adresu

Chybu a její co nejpřesnější popis, který může být doplněn o příslušný otisk obrazovky

Bez výše uvedených hodnot není možné hlášenou chybou zabývat

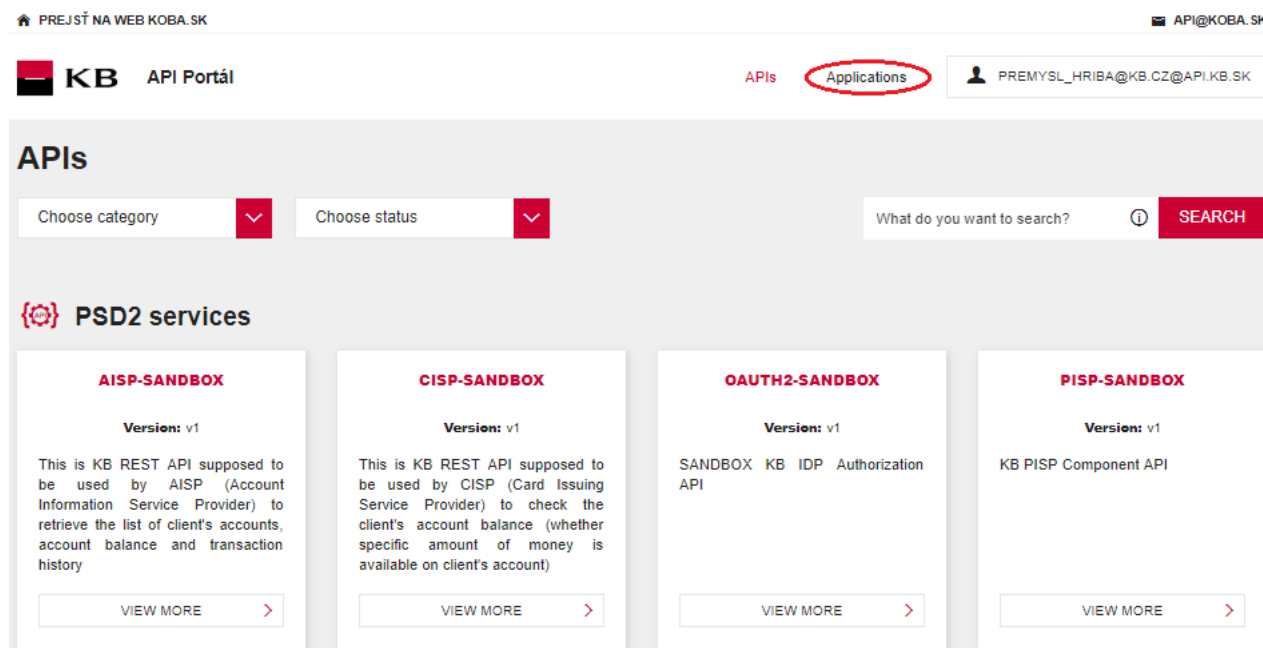
6. Postup pro vygenerování klíče/tokenu ke službě PISP

Předpoklady pro vygenerování klíče/tokenu

Uživatel je řádně zaregistrován a přihlášen do portálu Sandbox SK.

Vstup do aplikačního menu a výběr požadované aplikace

Přihlášený uživatel vstoupí do aplikačního menu pomocí odkazu Applications v horní částky obrazovky



The screenshot shows the KB API Portal interface. At the top, there is a navigation bar with the KB logo, 'API Portál', and a user profile dropdown for 'PREMYSL_HRIBA@KB.CZ@API.KB.SK'. A red circle highlights the 'Applications' link in the top navigation. Below the navigation bar, the main content area is titled 'APIs'. It features two dropdown menus for 'Choose category' and 'Choose status', and a search bar with the text 'What do you want to search?' and a 'SEARCH' button. Underneath, there is a section titled 'PSD2 services' with four service cards:







- AISP-SANDBOX** (Version: v1): This is KB REST API supposed to be used by AISP (Account Information Service Provider) to retrieve the list of client's accounts, account balance and transaction history.
- CISP-SANDBOX** (Version: v1): This is KB REST API supposed to be used by CISP (Card Issuing Service Provider) to check the client's account balance (whether specific amount of money is available on client's account).
- OAuth2-SANDBOX** (Version: v1): SANDBOX KB IDP Authorization API.
- PISP-SANDBOX** (Version: v1): KB PISP Component API.

Each card has a 'VIEW MORE' button with a right-pointing arrow.

[+ ADD APPLICATION](#)

Applications

An application is a logical collection of APIs. Applications allow you to use a single access token to invoke a collection of APIs and to subscribe to one API multiple times with different SLA levels. The DefaultApplication is pre-created and allows unlimited access by default.

Name	Tier	Workflow Status	Subscriptions	Actions
DefaultApplication	Unlimited	ACTIVE	0	  
Test_KOBA	Unlimited	ACTIVE	4	  

 Show entries Showing 1 to 2 of 2 entries

1

Výběr funkčnosti aplikačního menu

Uživatel v aplikačním menu zvolí položky „SANDBOX KEYS“

[< APPLICATION LIST](#)
[EDIT](#)

Test_KOBA

[DETAILS](#)
[SANDBOX KEYS](#)
[SUBSCRIPTIONS](#)

Status APPROVED

Unlimited Allows unlimited requests

Per Token Quota This feature allows you to assign an API request quota per access token. Allocated quota will be shared among all the subscribed APIs of the application.

Description Not Given

Generování certifikátu pro službu PISP

Uživatel v Sandboxu může zvolit vygenerování tokenu pro službu PISP za předpokladu, že je uživatel přihlášen k odběru služeb PISP.

Uživatel zvolí požadovaný scope pro generování klíče/tokenu a vygeneruje token pomocí funkčnosti „REGENERATE“

[< APPLICATION LIST](#)
[EDIT](#)

Test_KOBA

[DETAILS](#)
SANDBOX KEYS
[SUBSCRIPTIONS](#)
[SHOW KEYS](#)

Consumer Key

Consumer Secret

Grant Types

The application can use the following grant types to generate Access Tokens. Based on the application requirement, you can enable or disable grant types for this application.

 Refresh Token

 SAML2

 Implicit

 Password

 IWA-NTLM

 Client Credential

 Code

Callback URL

[UPDATE](#)

Generating Access Tokens

The following cURL command shows how to generate an access token using the Password Grant type.

```
curl -k -d "grant_type=password&username=Username&password=Password" \
-H "Authorization: Basic Base64(consumer-key:consumer-secret)" \
https://api.koba.sk/token
```

In a similar manner, you can generate an access token using the Client Credential grant type with the following cURL command.

```
curl -k -d "grant_type=client_credentials" \
-H "Authorization: Basic Base64(consumer-key:consumer-secret)" \
https://api.koba.sk/token
```

Generate a Test Access Token

Access Token

Above token has a validity period of 3600 seconds. If you want to regenerate this token, please select it's scopes and validity period.

Validity period

[REGENERATE](#)

7. Způsoby volání mocku PISP

1. [Volání mocku založení nové platby přes konzoli na API portálu](#)
2. [Volání mocku získání stavu platby přes konzoli na API portálu](#)
3. [Volání mocku PIS „Payment Authorization“ k otestování \(přes konzoli na API portálu\)](#)
4. [Přímým voláním resource iniciace platby](#)

5. [Přímým voláním resource získání stavu platby](#)

6. Přímým voláním resource autorizace platby - momentálně není na sandboxu implementováno.

8. Postup přístupu do aplikace přes konzoli na API portálu

Volání mocku PIS „New payment“ na otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „New payment“. Tato operace umožňuje založit novou platbu. Po kliknutí na „SHOW MORE“ se rozbalí daná operace.

PREJŠŤ NA WEB KOBA.SK API@KOBA.SK

KB API Portál APIs Applications PREMYSL_HRIBA@KB.CZ@API.KB.SK

[< GO BACK](#)

PISP-Sandbox

Version: v1 | Updated: 22/Jan/2020 12:05:50 PM CET | Status: PUBLISHED

KB PISP Component API

API CONSOLE | DOCUMENTATION

Try Using

Set Request Header
 Authorization : Bearer

[Swagger \(/swagger.json \)](#)

PISP

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

POST	New payment – payment initiation /payments	SHOW MORE
POST	Payment Authorization – starting the particular authorization method /payments/{paymentId}/sign/{signId}	SHOW MORE
GET	Payment status information /payments/{paymentId}/status	SHOW MORE
POST	Balance check /payments/balanceCheck	SHOW MORE

Applications: Tiers: [SUBSCRIBE](#)

Vyplnění požadovaných polí u operace „New payment“

Uživatel, který si přeje iniciovat novou platbu, vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl a neexistuje žádný jiný důvod proč platba nemohla proběhnout (např.: nedostatečný zůstatek na účtu), zobrazí se mu zpráva o provedení platby. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou rudě zvýrazněna. Pro „paymentRequest“ stačí kliknout na uvedený příklad a ten se zkopíruje do příslušného pole.

PREJŠT NA WEB KOBA.SK API@KOBA.SK

KB API Portál APIs Applications PREMYSL_HRIBA@KB.CZ@API.KB.SK

[GO BACK](#)

PISP-Sandbox

Version: v1 | Updated: 22/Jan/2020 12:05:50 PM CET | Status: PUBLISHED

KB PISP Component API

API CONSOLE DOCUMENTATION

Try Using Key

Set Request Header
 Authorization : Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.

[Swagger \(/swagger.json \)](#)

PISP

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#)

POST New payment – payment initiation /payments SHOW LESS ^

Required Scopes

Key	Name
pisp	pisp

Response Class (Status 200)
successful operation

Model | **Example Value**

```

{
  "paymentRequest": {
    "paymentIdentification": {
      "instructionIdentification": "NOTPROVIDED"
    },
    "paymentTypeInformation": {
      "instructionPriority": "NORM"
    },
    "amount": {
      "instructedAmount": {
        "value": "10.00",
        "currency": "EUR"
      }
    }
  }
}

```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
x-request-id	<input type="text"/>	External Request ID	header	string
TPP-Name	<input type="text" value="(required)"/>	Transaction initiator name	header	string
paymentRequest	<input type="text" value="(required)"/>	New payment request	body	

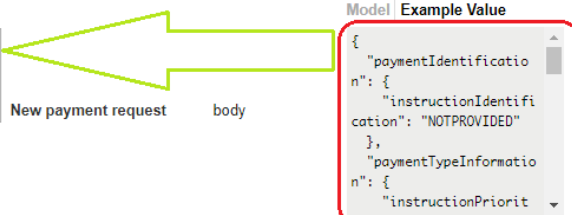
Parameter content type:

Model | **Example Value**

```

{
  "paymentIdentification": {
    "instructionIdentification": "NOTPROVIDED"
  },
  "paymentTypeInformation": {
    "instructionPriority": "NORM"
  }
}

```



Vypsání chyby u operace „New payment“

Pokud byla některá hodnota zadána špatně, vypíše se po zmáčknutí tlačítka „TRY IT OUT“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

Response Messages

HTTP Status Code	Reason	Response Model	Headers				
400	Input parameter is invalid	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>						
401	Missing certificate or access token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>						
403	Invalid certificate or token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>						
415	Invalid message charset	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>						
500	Unexpected error occurred	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>						


[TRY IT OUT](#)
[Hide Response](#)

Volání mocku PIS „Payment status information“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Payment status information“. Tato operace umožňuje zobrazit stav platby. Jedná se o založenou platbu, která ještě nebyla klientem autorizována, nebo již byla autorizována a PISP se dotazuje na její stav (GET). Po kliknutí na „SHOW MORE“ se rozbálí daná operace.


🏠 PREJŠŤ NA WEB KOBA.SK

✉ API@KOBA.SK



API Portál

APIs
Applications


PREMYSL_HRIBA@KB.CZ@API.KB.SK

[< GO BACK](#)

PISP-Sandbox

Version: v1
Updated: 22/Jan/2020 12:05:50 PM CET
Status: PUBLISHED

KB PISP Component API

API CONSOLE

DOCUMENTATION

Try TEST_KOBA ▼ Using SANDBOX ▼ Key

Set Request Header

Authorization : Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.

[Swagger \(/swagger.json \)](#)

PISP Show/Hide | List Operations | Expand Operations


POST	New payment – payment initiation <code>/payments</code>	SHOW MORE ▼
POST	Payment Authorization – starting the particular authorization method <code>/payments/{paymentId}/sign/{signId}</code>	SHOW MORE ▼
GET	Payment status information <code>/payments/{paymentId}/status</code>	SHOW MORE ▼
POST	Balance check <code>/payments/balanceCheck</code>	SHOW MORE ▼

Applications

DEFAULTAPPLICATION
▼

Tiers

UNLIMITED
▼

SUBSCRIBE


Vyplnění požadovaných polí u operace „Payment status information“

Uživatel, který si přeje zobrazit stav určité platby, vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl, zobrazí se mu informace o dané platbě. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou rudě zvýrazněna.

GET

Payment status information
/payments/{paymentId}/status

SHOW LESS ^

Required Scopes

Key	Name
pisp	pisp

Response Class (Status 200)
successful operation

Model **Example Value**

```

{
  "instructionStatus": "ACTC",
  "errorInfo": {
    "error": "string",
    "parameters": {},
    "scope": "string",
    "message": "string"
  }
}

```

Response Content Type v

Parameters

Parameter	Value	Description	Parameter Type	Data Type
x-request-id	<input type="text"/>	External Request ID	header	string
TPP-Name	<input type="text"/>	Transaction initiator name	header	string
paymentId	<input type="text" value="(required)"/>	Unique bank transaction identification • example: 'ZU000007AX3'	path	string

Vypsání chyby u operace „Payment status information“

Pokud byla některá hodnota zadána špatně, vypíše se po zmáčknutí tlačítka „TRY IT OUT“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

Response Messages

HTTP Status Code	Reason	Response Model	Headers				
400	Input parameter is invalid	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>						
401	Missing certificate or access token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>						
403	Invalid certificate or token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>						
404	Id does not exist	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>						
415	Invalid message charset	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>						
500	Unexpected error occurred	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>						

[TRY IT OUT](#)
[Hide Response](#)

Volání mocku PIS „Payment Authorization“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Payment Authorization“. Tato operace umožňuje nastartovat konkrétní autorizační metody. Po kliknutí na „SHOW MORE“ se rozbalí daná operace.

PREJŠŤ NA WEB KOBA.SK

API@KOBA.SK

KB API Portál

APIs

Applications

PREMYSL_HRIBA@KB.CZ@API.KB.SK

< GO BACK

PISP-Sandbox

Version: v1

Updated: 22/Jan/2020 12:05:50 PM CET

Status: PUBLISHED

KB PISP Component API

API CONSOLE

DOCUMENTATION

Try TEST_KOBA

Using SANDBOX

Key

Set Request Header

Authorization : Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.

Swagger (/swagger.json)

PISP

Show/Hide | List Operations | Expand Operations

POST

New payment – payment initiation
/payments

SHOW MORE

POST

Payment Authorization – starting the particular authorization method
/payments/{paymentId}/sign/{signId}

SHOW MORE

GET

Payment status information
/payments/{paymentId}/status

SHOW MORE

POST

Balance check
/payments/balanceCheck

SHOW MORE

Applications

DEFAULTAPPLICATION

Tiers

UNLIMITED

SUBSCRIBE

Vyplnění požadovaných polí u operace „Payment Authorization“

Uživatel, který si přeje nastartovat autorizační proces, vyplní všechna pole hodnotami v příslušném formátu. Pokud vše provedl, jak měl, zobrazí se mu přehled hodnot potřebných pro dokončení autorizace. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněná pole jsou rudě zvýrazněna. Pro pole „authInitiationRequest“ stačí kliknout na uvedený příklad a ten se zkopíruje do příslušného pole.

POST

Payment Authorization – starting the particular authorization method

/payments/{paymentId}/sign/{signId}

SHOW LESS ^

🕒 **Required Scopes**

Key	Name
pisp	pisp

Response Class (Status 200)

successful operation

Model | Example Value

```

},
"method": "string",
"formData": {
  "SAMLRequest": "string",
  "relayState": "string"
},
"signInfo": {
  "state": "OPEN",
  "signId": "string",
  "signInfo": "ACTC"
}
}
                
```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
x-request-id	<input type="text"/>	External Request ID	header	string
TPP-Name	<input type="text"/>	Transaction initiator name	header	string
paymentId	<input type="text" value="ZU000007AX3"/>	Unique bank transaction identification example: 'ZU000007AX3'	path	string
signId	<input type="text" value="ZU000007AX3"/>	Identifier of the change	path	string
authInitiationRequest	<div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;">(required)</div>	Initiation request identifier of the authorization method and back URL	body	string

Parameter content type: application/json

Model | Example Value

```

{
  "authorizationType": "USER_AGENT_REDIRECT",
  "backUrl": "string"
}
                
```


Vypsání chyby u operace „Payment Authorization“

Pokud byla některá hodnota zadána špatně, vypíše se po zmáčknutí tlačítka „TRY IT OUT“ jedna z následujících chybových hlášek, popřípadě error specifikovaný v definici mocku, jinak proběhne výpis výsledku.

Response Messages

HTTP Status Code	Reason	Response Model	Headers				
400	Input parameter is invalid	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_400", "scope": "x-request-id", "message": "Value of parameter x-request-id is wrong" }] }</pre>						
401	Missing certificate or access token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_401", "message": "Missing certificate or access token" }] }</pre>						
403	Invalid certificate or token	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_403", "message": "Invalid certificate or token" }] }</pre>						
404	Id does not exist	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ID_NOT_FOUND", "message": "Parameter paymentId not found" }] }</pre>						
415	Invalid message charset	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "RR10", "message": "InvalidCharacterSet" }] }</pre>						
500	Unexpected error occurred	<table border="1"> <thead> <tr> <th>Model</th> <th>Example Value</th> </tr> </thead> <tbody> <tr> <td></td> <td> <pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre> </td> </tr> </tbody> </table>	Model	Example Value		<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>	
Model	Example Value						
	<pre>{ "errors": [{ "error": "ERR_CODE_500", "message": "Internal Server Error" }] }</pre>						

[TRY IT OUT](#)
[Hide Response](#)

Volání mocku PIS „Balance check“ k otestování

Uživatel vybere operaci, kterou si přeje otestovat. V tomto případě vybere operaci „Balance Check“. Tato operace umožňuje získat informaci o dostupnosti prostředků na účtů klienta.

PREJSTĚ NA WEB KOBA.SK API@KOBA.SK

KB API Portál APIs Applications PREMYSL_HRIBA@KB.CZ@API.KB.SK

[< GO BACK](#)

PISP-Sandbox

Version: v1 | Updated: 22/Jan/2020 12:05:50 PM CET | Status: PUBLISHED

KB PISP Component API

API CONSOLE DOCUMENTATION

Try Using Key

Set Request Header

Authorization : Bearer Swagger (/swagger.json)

PISP Show/Hide | List Operations | Expand Operations

POST	New payment – payment initiation /payments	SHOW MORE ▾
POST	Payment Authorization – starting the particular authorization method /payments/{paymentId}/sign/{signId}	SHOW MORE ▾
GET	Payment status information /payments/{paymentId}/status	SHOW MORE ▾
POST	Balance check /payments/balanceCheck	SHOW MORE ▾

Applications Tiers SUBSCRIBE

Vyplnění požadovaných polí u operace „Check Balance“

Pro requestBody stačí kliknout na uvedený příklad a ten se zkopíruje do příslušného pole. V případě, že nejsou vyplněna všechna povinná pole, výpis se neprovede a nevyplněné pole je červeně zvýrazněno

POST
Balance check
/payments/balanceCheck
SHOW LESS ^

🕒 **Required Scopes**

Key	Name
pisp	pisp

Response Class (Status 200)

successful operation

Model | **Example Value**

```
{
  "responseIdentification": 987654,
  "exchangeIdentification": "103149078",
  "response": "APPR"
}
```

Response Content Type ▼

Parameters

Parameter	Value	Description	Parameter Type	Data Type
x-request-id	<input type="text"/>	External Request ID	header	string
TPP-Name	<input type="text" value="(required)"/>	Transaction initiator name	header	string

balanceCheckRequest

Parameter content type: ▼

Query for Balance Check

body

Model | **Example Value**

```
{
  "exchangeIdentification": "103149078",
  "card": {
    "cardHolderName": "Jan Novák",
    "maskedPAN": "1234**
    *****6789"
  },
}
```

9. Postup přístupu do aplikace přes přímé volání

Nová platba – I. iniciace platby (POST /my/payments)

Resource pro založení nové platby.

Charakteristika resource

URI: /payments
HTTP Metoda: POST
Request URL: <https://api.koba.sk/sandbox/pisp/v1/payments>
Authorization: request **vyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne

Třídění: ne

Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specification of required transfer format. From the precondition of technical specification of this API standard, in this case, application/json format is primarily supported.

Obsah POST requestu a response volání viz kapitola 0 ELEMENTY ZPRÁVY Nová platba – iniciace platby

Tabulka - Chybové kódy definované ČBA standardem pro službu POST iniciace platby

HTTP STATUS KÓD	ERROR KÓD	ÚČEL
401	UNAUTHORISED	Chybějící certifikát.
403	FORBIDDEN	Volání metody, která neodpovídá licenci, nebo neplatný certifikát.
400	FIELD_MISSING	Chybějící povinné pole v requestu.
400	FIELD_INVALID	Hodnota pole není validní.
400	AC02	[InvalidDebtorAccountNumber] – nevalidní identifikátoru účtu v obsahu požadavku.

400	AC03	[InvalidCreditorAccountNumber] - číslo účtu příjemce je uzavřeno, blokováno, příp. pro daný typ účtu jsou zakázány kreditní transakce nebo je číslo účtu příjemce uvedeno v nevalidním formátu (poznámka: validováno pouze pro in-house platby).
400	AC10	[InvalidDebtorAccountCurrency] – uvedená měna účtu plátce neodpovídá měně účtu klienta pro dané číslo účtu vedené v bance (měna účtu je nepovinná, v případě multicurrency účtů je měna účtu nutné specifikovat - Raiffeisenbank).
400	AC12	[InvalidAccountType] - typ účtu neodpovídá povoleným typům účtů (např. se jedná o neplatební účet).
403	AG01	[TransactionForbidden] – neexistující souhlas s přístupem k informaci o dostatku prostředků na účtu.
400	AM05	[Duplication] - jedná se o duplicitu. Univerzální kód pro duplicitní dotaz CISP (validuje například ČSOB) nebo se jedná o duplicitní platbu přes PISP (neunikátní reference platby).
400	AM11	[InvalidTransactionCurrency] – v požadavku je uvedena neobchodovaná/nepodporovaná měna.
400	AM12	[InvalidAmount] – chybně zadaná částka. Např. příliš nízká, nebo vysoká částka, či špatný formát čísla včetně počtu desetinných míst dle ISO 4217.
400	FF01	[Invalid File Format] – nevalidní JSON formát, či jiný technický problém se zpracováním dotazu.
400	BE19	[InvalidChargeBearerCode] - neplatný typ poplatku pro daný typ transakce.
400	DT01	[InvalidDate] - "Datum neodpovídá" – viz. Níže *
400, 50x	NARR	Narrative – obecný důvod pro odmítnutí platby, s doplněním informace o chybě.
400	RC07	[InvalidCreditorBICIdentifier] – neplatný SWIFT / BIC kód banky příjemce.
400	RC10	[InvalidCreditorClearingSystemMemberIdentifier] - neplatná identifikace kódu banky příjemce.
400	RR03	[MissingCreditorNameOrAddress] - v poli chybí požadovaná data o názvu či adrese (nebo její části) příjemce. Pokud jsou údaje ve špatném formátu, je využitý error code FIELD_INVALID
400	RR10	[InvalidCharacterSet] – nevalidní znaková sada v požadavku.

Elementy zprávy Nová platba – iniciace platby

Uvažované typy plateb

KÓD PLATBY	SERVICELLEVELCODE	POPIS
SEPA	ESCT	SEPA platba (domácí a v rámci EHP)
ZPL	XBCT	Zahraniční platba v rámci EHP, Zahraniční platba mimo EHP

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZENTACE
+	paymentIdentification	[1..1]	ALL	PaymentIdentification1	Identifikace platby

++	instructionIdentification	[1..1]	ALL	Max35Text	Identifikace instrukce
++	endToEndIdentification	[0..0] [1..1] [0..0] [0..0]	SEPA ZPL	Max35Text	Identifikace End To End.
++	transactionIdentification	[0..0]	ALL	Max35Text	Identifikace transakce
+	paymentTypeInformation	[0..1]	ALL	PaymentTypeInformation19	Informace o typu platby
++	instructionPriority	[0..1]	ALL	Priority2Code	Priorita instrukce
++	serviceLevel	[0..0]	ALL	ServiceLevel8CZ	Úroveň služeb
+++	code	[0..0]	ALL	ExternalServiceLevel1Code	Kód úrovně služeb
++	categoryPurpose	[0..0]	ALL	CategoryPurpose1Choice	Kategorie účelu platby
+++	code	[0..0]	ALL	ExternalCategoryPurpose1Code	Kód kategorie účelu platby
+++	proprietary	[0..0]	ALL	Max35Text	Volný formát kategorie účelu platby
+	amount	[1..1]	ALL	SEPA- AmountType3CZ EHP- AmountType3Choice NONEHP- AmountType3Choice	Částka
++	instructedAmount	[1..1]	ALL	CurrencyAndAmount	Částka a měna v instrukci
+++	value	[1..1]	ALL	Amount	Částka převodu
+++	currency	[1..1]	ALL	CurrencyCode	Měna převodu
++	equivalentAmount	[0..0]	ALL	CurrencyAndAmount	Ekvivalentní částka a měna
+++	value	[0..0]	ALL	Amount	Ekvivalentní částka transakce
+++	currency	[0..0]	ALL	CurrencyCode	Měna ekvivalentní částky transakce
+	requestedExecutionDate	[0..1]	ALL	ISODate	Požadované datum provedení platby
+	exchangeRateInformation	[0..0]	ALL	ExchangeRateInformation1	Smluvní kurz
++	exchangeRate	[0..0]	ALL	BaseOneRate	Dohodnutý směnný kurz
++	rateType	[0..0]	ALL	ExchangeRateType1Code	Typ dohodnutého směnného kurzu
++	contractIdentification	[0..0]	ALL	Max35Text	Identifikátor použití dohodnutého směnného kurzu
+	chargeBearer	[0..0] [0..0] [0..1]	SEPA ZPL	ChargeBearerType1Code	Plátce poplatků
+	chargesAccount	[0..0]	ALL	CashAccount16CZ	Účet pro poplatky
++	identification	[0..0]	ALL	AccountIdentification4ChoiceCZ	Identifikace čísla účtu pro poplatky
+++	iban	[0..0]	ALL	IBAN2007Identifier	Číslo účtu pro poplatky ve formátu IBAN
++	currency	[0..0]	ALL	CurrencyCode ISO 4217	Měna účtu pro poplatky
+	ultimateDebtor	[0..0] [0..1] [0..0]	SEPA ZPL	PartyIdentification32CZ1	Původní plátce

++	name	[0..0] [0..1] [0..0]	SEPA ZPL	Max70Text	Jméno původního plátce
++	postalAddress	[0..0] [0..1] [0..0]	SEPA ZPL	PostalAddress6CZ	Jméno původního plátce
+++	streetName	[0..0] [0..1] [0..0]	SEPA ZPL	Max70Text	Ulice původního plátce
+++	buildingNumber	[0..0] [0..1] [0..0]	SEPA ZPL	Max16Text	Číslo budovy původního plátce
+++	postCode	[0..0] [0..1] [0..0]	SEPA ZPL	Max16Text	PSČ původního plátce
+++	townName	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Město původního plátce
+++	country	[0..0] [0..1] [0..0]	SEPA ZPL	CountryCode ISO3166	Země původního plátce
+++	addressLine	[0..0] [0..2] [0..0]	SEPA ZPL	Max70Text	Nestrukturovaný zápis adresy původního plátce.
++	identification	[0..0] [0..1] [0..0]	SEPA ZPL	Party6Choice	Identifikace původního plátce.
+++	organisationIdentification	[0..0] [1..1] [0..0]	SEPA ZPL	OrganisationIdentification 4CZ	Jméno původního plátce
++++	bicOrBei	[0..0] [0..1] [0..0]	SEPA ZPL	BICIdentifier	Identifikace původního plátce jako organizace/právnícké osoby ve formě BIC nebo BEI kódu.

++++	other	[0..0] [0..1] [0..0]	SEPA ZPL	GenericOrganisationIdent ification1	Jiná identifikace původního plátce jako organizace/právnícké osoby.
+++++	identification	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Jiná identifikace původního plátce jako organizace/právnícké osoby v nestrukturované podobě.
+++++	schemeName	[0..0] [0..1] [0..0]	SEPA ZPL	OrganisationIdentification SchemeName1CZ	Typ dokumentu pro identifikaci původního plátce jako organizace/právnícké osoby.
+++++	proprietary	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Typ dokumentu pro identifikaci původního plátce jako organizace/právnícké osoby ve volném textovém formátu.
+++++	issuer	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Výstavce dokumentu pro identifikaci původního plátce jako organizace/právnícké osoby.
+++	privateIdentification	[0..0] [1..1] [0..0]	SEPA ZPL	PersonIdentification5CZ	Jednoznačná identifikace původního plátce jako fyzické osoby. buď organisationIdentification nebo privateIdentification.
++++	other	[0..0] [0..1] [0..0]	SEPA ZPL	GenericPersonIdentificati on1	Jiná identifikace původního plátce jako fyzické osoby v nestrukturované podobě.
+++++	identification	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Jiná identifikace původního plátce jako fyzické osoby v nestrukturované podobě.
+++++	schemeName	[0..0] [0..1] [0..0]	SEPA ZPL	PersonIdentificationSche meName1Choice	Typ dokumentu pro identifikaci původního plátce jako fyzické osoby.
+++++	proprietary	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Typ dokumentu pro identifikaci původního plátce jako fyzické osoby ve volném textovém formátu.
+++++	issuer	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Výstavce dokumentu pro identifikaci původního plátce jako fyzické osoby.
+	debtor	[0..0]	ALL	PartyIdentification32CZ2	Plátce
++	name	[0..0]	ALL	Max70Text	Jméno plátce

++	postalAddress	[0..0]	ALL	PostalAddress6CZ	Poštovní adresa plátce
+++	streetName	[0..0]	ALL	Max70Text	Název ulice použitý pro poštovní adresu plátce.
+++	buildingNumber	[0..0]	ALL	Max16Text	Číslo popisné použité pro poštovní adresu plátce.
+++	postCode	[0..0]	ALL	Max16Text	Poštovní směrovací číslo použité pro poštovní adresu plátce.
+++	townName	[0..0]	ALL	Max35Text	Název města použitý pro poštovní adresu plátce.
+++	country	[0..0]	ALL	CountryCode ISO3166	Název země použitý pro poštovní adresu plátce.
+++	addressLine	[0..0]	ALL	Max70Text	Nestrukturovaný záznam poštovní adresy plátce.
+	debtorAccount	[1..1]	ALL	CashAccount16CZ	Účet plátce
++	identification	[1..1]	ALL	AccountIdentification4ChoiceCZ	Identifikace účtu plátce
+++	iban	[1..1]	ALL	IBAN2007Identifier	Číslo účtu plátce ve formátu IBAN
+++	other	[0..0]	ALL	GenericAccountIdentification1CZ	Jiný formát čísla účtu plátce
++++	identification	[0..0]	ALL	Max34Text	Číslo účtu plátce v lokálním formátu BBAN
++	currency	[0..1]	ALL	CurrencyCode ISO 4217	Měna účtu plátce
+	intermediaryAgent1	[0..0]	ALL	BranchAndFinancialInstitutionIdentification4CZ	Mezibanka 1
++	financialInstitutionIdentification	[0..0]	ALL	FinancialInstitutionIdentification7CZ	Identifikace finanční instituce
+++	bic	[0..0]	ALL	BICIdentifier	BIC / SWIFT kód banky
+++	clearingSystemMemberIdentification	[0..0]	ALL	ClearingSystemMemberIdentification2	Identifikace účastníka clearingového systému
++++	clearingSystemIdentification	[0..0]	ALL	ClearingSystemIdentification2Choice	Identifikace clearingového systému
+++++	code	[0..0]	ALL	ClearingSystemIdentification1Code	Kód
+++++	proprietary	[0..0]	ALL	Max35Text	Volný formát
++++	memberIdentification	[0..0]	ALL	Max35Text	Clearingový kód účastníka
+++	name	[0..0]	ALL	Max70Text	Jméno
+++	postalAddress	[0..0]	ALL	PostalAddress6CZ	Poštovní adresa
++++	streetName	[0..0]	ALL	Max70Text	Ulice
++++	buildingNumber	[0..0]	ALL	Max16Text	Číslo budovy
++++	postCode	[0..0]	ALL	Max16Text	PSC
++++	townName	[0..0]	ALL	Max35Text	Město
++++	country	[0..0]	ALL	CountryCode ISO3166	Země
++++	addressLine	[0..0]	ALL	Max70Text	Nestrukturovaný zápis adresy
+++	other	[0..0]	ALL	GenericFinancialIdentification1CZ	Jiná identifikace banky
++++	identification	[0..0]	ALL	Max35Text	Lokální kód banky
+	creditorAgent	[0..0] [0..1] [1..1]	SEPA ZPL	BranchAndFinancialInstitutionIdentification4CZ	Banka příjemce.

++	financialinstitutionidentification	[0..0] [0..1] [1..1]	SEPA ZPL	FinancialInstitutionIdentification7CZ	Identifikace finanční instituce.
+++	bic	[0..0] [1..1] [0..1]	SEPA ZPL	BICIdentifier	BIC / SWIFT kód banky.
+++	clearingSystemMemberIdentification	[0..0] [0..0] [0..1]	SEPA ZPL	ClearingSystemMemberIdentification2	Identifikace účastníka clearingového systému.
++++	clearingSystemIdentification	[0..0] [0..0] [0..1] ... buď anebo některá z jiných variant identifikace banky příjemce	SEPA ZPL	ClearingSystemIdentification2Choice	Identifikace clearingového systému.
+++++	code	[0..0] [0..0] [1..1]... buď code anebo proprietary	SEPA ZPL	ExternalClearingSystemIdentification1Code	Kód.
+++++	proprietary	[0..0] [0..0] [1..1]... buď code anebo proprietary	SEPA ZPL	Max35Text	Volný formát.
++++	memberIdentification	[0..0] [0..0] [1..1]	SEPA ZPL	Max35Text	Clearingový kód účastníka.
+++	name	[0..0] [0..0] [0..1]... buď anebo některá z jiných variant identifikace banky příjemce	SEPA ZPL	Max70Text	Jméno.
+++	postalAddress	[0..0] [0..0] [0..1]... buď anebo některá z jiných variant identifika	SEPA ZPL	PostalAddress6CZ	Poštovní adresa.

		ce banky příjemce			
++++	streetName	[0..0] [0..0] [0..1]	SEPA ZPL	Max70Text	Ulice. .
++++	buildingNumber	[0..0] [0..0] [0..1]	SEPA ZPL	Max16Text	Číslo budovy.
++++	postCode	[0..0] [0..0] [0..1]	SEPA ZPL	Max16Text	PSC.
++++	townName	[0..0] [0..0] [0..1]	SEPA ZPL	Max35Text	Město.
++++	country	[0..0] [0..0] [0..1]	SEPA ZPL	CountryCode ISO3166	Země.
++++	addressLine	[0..0] [0..0] [0..2]	SEPA ZPL	Max70Text	Nestrukturovaný zápis adresy.
+++	other	[0..0] [0..0] [0..1]... buď anebo některá z jiných variant identifika ce banky příjemce	SEPA ZPL	GenericFinancialIdentific ation1C	Jiná identifikace banky.
++++	identification	[0..0] [0..0] [0..1]	SEPA ZPL	Max35Text	Lokální kód banky.
+	creditor	[0..0] [1..1] [1..1]	SEPA ZPL	PartyIdentification32CZ2	Příjemce.

++	name	[0..0] [1..1] [1..1]	SEPA ZPL	Max70Text	Jméno příjemce.
++	postalAddress	[0..0] [0..1] [1..1]	SEPA ZPL	PostalAddress6CZ	Poštovní adresa.
+++	streetName	[0..0] [0..1] [0..1]	SEPA ZPL	Max70Text	Ulice.
+++	buildingNumber	[0..0] [0..1] [0..1]	SEPA ZPL	Max16Text	Číslo budovy.
+++	postCode	[0..0] [0..1] [0..1]	SEPA ZPL	Max16Text	PSC.
+++	townName	[0..0] [0..1] [0..1]	SEPA ZPL	Max35Text	Město.
+++	country	[0..0] [0..1] [0..1]	SEPA ZPL	CountryCode ISO3166	Země.
+++	addressLine	[0..0] [0..2] [0..2]	SEPA ZPL	Max70Text	Nestrukturovaný zápis adresy.
+	creditorAccount	[1..1]	ALL	CashAccount16CZ	Účet příjemce
++	identification	[1..1]	ALL	AccountIdentification4Ch oiceCZ	Identifikace účtu příjemce
+++	iban	[1..1]	ALL	IBAN2007Identifier	Číslo účtu ve formátu IBAN
+++	other	[0..0] [0..0] [1..1]	SEPA ZPL	GenericAccountIdentificat ion1CZ:	Jiný formát čísla účtu.
++++	identification	[0..0] [0..0] [1..1]	SEPA ZPL	Max34Text	Číslo účtu v lokálním formátu BBAN.

++	currency	[0..1] [0..0] [0..0]	SEPA ZPL	CurrencyCode ISO4217	Měna účtu příjemce.
+	ultimateCreditor	[0..0] [0..1] [0..0]	SEPA ZPL	PartyIdentification32CZ1	Konečný příjemce.
++	name	[0..0] [0..1] [0..0]	SEPA ZPL	Max70Text	Jméno.
++	postalAddress	[0..0] [0..1] [0..0]	SEPA ZPL	PostalAddress6CZ	Poštovní adresa.
+++	streetName	[0..0] [0..1] [0..0]	SEPA ZPL	Max70Text	Ulice.
+++	buildingNumber	[0..0] [0..1] [0..0]	SEPA ZPL	Max16Text	Číslo budovy.
+++	postCode	[0..0] [0..1] [0..0]	SEPA ZPL	Max16Text	PŠČ
+++	townName	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Město
+++	country	[0..0] [0..1] [0..0]	SEPA ZPL	CountryCode ISO3166	Země
+++	addressLine	[0..0] [0..2] [0..0]	SEPA ZPL	Max70Text	Nestrukturovaný zápis adresy
++	identification	[0..0] [0..1] [0..0]	SEPA ZPL	Party6Choice	Identifikace konečného příjemce

+++	organisationIdentification	[0..0] [1..1] [0..0]	SEPA ZPL	OrganisationIdentification4CZ	Jednoznačná identifikace konečného příjemce jako organizace/právnícké osoby. buď organisationIdentification nebo privateIdentification.
++++	bicOrBei	[0..0] [0..1] [0..0]	SEPA ZPL	BICIdentifier	Identifikace konečného příjemce jako organizace/právnícké osoby ve formě BIC nebo BEI kódu.
++++	other	[0..0] [0..1] [0..0]	SEPA ZPL	GenericOrganisationIdentification1	Jiná identifikace konečného příjemce jako organizace/právnícké osoby.
+++++	identification	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Jiná identifikace konečného příjemce jako organizace/právnícké osoby v nestrukturované podobě.
+++++	schemeName	[0..0] [0..1] [0..0]	SEPA ZPL	OrganisationIdentificationSchemeName1CZ	Typ dokumentu pro identifikaci konečného příjemce jako organizace/právnícké osoby.
+++++	proprietary	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Typ dokumentu pro identifikaci konečného příjemce jako organizace/právnícké osoby ve volném textovém formátu.
+++++	issuer	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Výstavce dokumentu pro identifikaci konečného příjemce jako organizace/právnícké osoby.
+++	privateIdentification	[0..0] [1..1] [0..0]	SEPA ZPL	PersonIdentification5CZ	Jednoznačná identifikace konečného příjemce jako fyzické osoby. buď organisationIdentification nebo privateIdentification.
++++	other	[0..0] [0..1] [0..0]	SEPA ZPL	GenericPersonIdentification1	Jiná identifikace konečného příjemce jako fyzické osoby v nestrukturované podobě.
+++++	identification	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Jiná identifikace konečného příjemce jako fyzické osoby v nestrukturované podobě.
+++++	schemeName	[0..0] [0..1] [0..0]	SEPA ZPL	PersonIdentificationSchemeName1Choice	Typ dokumentu pro identifikaci konečného příjemce jako fyzické osoby.

+++++	proprietary	[0..0] [1..1] [0..0]	SEPA ZPL	Max35Text	Typ dokumentu pro identifikaci konečného příjemce jako fyzické osoby ve volném textovém formátu.
+++++	issuer	[0..0] [0..1] [0..0]	SEPA ZPL	Max35Text	Výstavce dokumentu pro identifikaci konečného příjemce jako fyzické osoby.
+	purpose	[0..0]	ALL	Purpose2Choice	Účel platby
++	code	[0..0]	ALL	ExternalPurpose1Code	Kód účelu platby
++	proprietary	[0..0]	ALL	Max35Text	Volný formát účelu platby
+	instructionForNextAgent	[0..0]	ALL	Instruction code	Instrukce pro další banku
+	remittanceInformation	[0..1]	ALL	RemittanceInformation5CZ	Informace o platbě
++	unstructured	[0..1]	ALL	Max140Text, alfanumerických znaků podporovaných CERTIS (clearing ČNB), a to včetně podporovaných speciálních znaků	Nestrukturovaná zpráva pro příjemce (viz. Níže *)
++	structured	[0..1] [0..0] [0..0]	SEPA ZPL	StructuredRemittanceInformation7CZ	Strukturovaná zpráva pro příjemce – variabilní, specifický a konstantní symbol
+++	creditorReferenceInformation	[0..1] [0..0] [0..0]	SEPA ZPL	CreditorReferenceInformation2CZ	Informace o referenci příjemce
++++	reference	[0..3] [0..0] [0..0]	SEPA ZPL	CreditorReferenceInformation2CZ	Hodnota VS, SS, KS

* **remittanceInformation.unstructured** - (tento údaj není povinný): používá se pro bližší identifikaci platby. Na Slovensku se k tomuto účelu používají platební symboly (variabilní, specifický a konstantní symbol). Jsou přenášeny též v poli „**Referencia platiteľa**“ (**EndToEndIdentification**) ve smyslu národně dohodnutých pravidel.

Pozn. k poli **remittanceInformation.unstructured**: dle standardu České bankovní asociace může i toto pole obsahovat informace o VS, KS a SS, nicméně Banka bude s veškerými informacemi zde obsaženými pracovat tak, jako by se jednalo o prostý popis platby (informace pro příjemce) a k žádnému parsování symbolů nebude docházet, ani kdyby se zde vyskytly.

- Hodnota **variabilního symbolu** se zaznamenává jako VS:max.10 číslic (např. VS:3451859072).
- Hodnota **konstantního symbolu** se zaznamenává jako KS:max.10 znaků (např. KS:0308).
- Hodnota **specifického symbolu** se zaznamenává jako SS:max.10 číslic (např. SS:8451201274).

JSON příklad elementu: "Referencia platiteľa":

Forma zadání symbolů dle slovenské národní konvence není libovolná, ale je předepsaná takto: **VS1234567890/SS1234567890/KS1234** **anebo** **VS1234567890/SS/KS1234**

Rovněž platí tato podmínka:

V případě zahraniční SEPA platby pole „Referencia platiteľa“:

1. nesmí obsahovat znak „/“ na první nebo poslední pozici,

2. nesmí obsahovat „/“ (dvě lomítka za sebou).

Dále upozorňujeme, že Banka ve zpracování AISP informací veškerá lomítka zadaná v PISP v poli **Remittance Information.Unstructured** nevyhodnocuje jako validní, resp. významová a v AISP výstupu je pak už neuvádí.

Například:

„VS: 36718//“, se u platby v AISP zapíše takto „VS: 36718“, tzn., že všechna lomítka systém zahodí.

Elementy response Nová platba - iniciace platby

Tabulka obsahuje jen elementy, které se vyskytují pouze v response zprávě.

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	transactionIdentification	[1..1]	Max35Text	Identifikátor založené transakce
+	serviceLevel	[1..1]	±	Zařazení služby (v rámci typu platby)
++	code	[1..1]	Text	Typ zadané platby
+	signInfo	[1..1]	±	Informace o stavu a id neautorizované transakce
++	state	[1..1]	StateCode	Informace o stavu autorizace transakce
++	signId	[0..1]	Text	Identifikátor autorizačního procesu konkrétní transakce.
++	signInfo	[1..1]	Status Code set	Identifikátor stavu transakce.

Hodnoty elementu serviceLevel.code – Typ iniciované platby

Kód	POPIS
DMCT	[DoMestic Credit Transfer] Domestic payment. Není podporováno pro SK.
ESCT	[SEPA Credit Transfer] – SEPA payment
XBCT	[Cross-Border Credit Transfer] – Cross border payment

Status kódy plateb – StatusCode

HTTP STATUS KÓD	STATUSCODE	ÚČEL
200	ACTC	[AcceptedTechnicalValidation] - Authentication and syntactical and semantical validation are successful
200	RJCT	[Rejected] - Payment initiation or individual transaction included in the payment initiation has been rejected
200	ACWC	[AcceptedWithChange] - Instruction is accepted but a change will be made, such as date or remittance not change

Status založené/iniciované platby (GET /payments/{paymentId}/status)

Resource pro zobrazení stavu platby. Jedná se o založenou platbu, která ještě nebyla klientem autorizována, nebo již byla autorizována a PISP se dotazuje na její stav (GET).

Resource vrací informace jen k transakcím založeným prostřednictvím konkrétního providera. Informace o providerovi je přebírána z certifikátu, resp. z informace o typu licence.

Uživatelská autorizace tohoto resource je optional. Primárně je vyžadován pouze platný certifikát providera.

Charakteristika resource

URI: /payments/{paymentId}/status
HTTP Metoda: GET
Request URL: <https://api.koba.sk/sandbox/pisp/v1/payments/{paymentId}/status>
Authorization: request **nevyžaduje** autorizaci uživatele/klienta jako součást volání API
Certification: request **vyžaduje** použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu.

Stránkování: ne
Třídění: ne
Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydany komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Obsah POST requestu a response volání viz kapitola 0 ELEMENTY ZPRÁVY Status založené/iniciované platby

Chybové kódy definované ČBA standardem pro službu GET Status založené/iniciované platby

HTTP STATUS KÓD	ERROR KÓD	ÚČEL
401	UNAUTHORISED	Nevalidní/chybějící certifikát = provider není autentizován
404	TRANSACTION_MISSING	Volání metody, která neodpovídá licenci, nebo neplatný certifikát.

Elementy zprávy Status založené/iniciované platby

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZENTACE
+	instructionStatus	[1..1]	PISP ALL	StatusCode	Status založené platby

Status kódy plateb – StatusCode

HTTP STATUS KÓD	STATUSCODE	ÚČEL
200	ACTC	[AcceptedTechnicalValidation] - Authentication and syntactical and semantical validation are successful

200	RJCT	[Rejected] - Payment initiation or individual transaction included in the payment initiation has been rejected
200	ACSP	[AcceptedSettlementInProgress] - All preceding checks such as technical validation and customer profile were successful and therefore the payment initiation has been accepted for execution
200	ACSC	[AcceptedSettlementCompleted] - Settlement on the debtor's account has been completed. Usage: this can be used by the first agent to report to the debtor that the transaction has been completed. Warning: this status is provided for transaction status reasons, not for financial information. It can only be used after bilateral agreement
200	ACWC	[AcceptedWithChange] - Instruction is accepted but a change will be made, such as date or remittance not change

Krok II. Inicie autorizace platby – specifické pro každou banku (POST /my/payments/{paymentId}/sign/{signId})

Tento resource je určen ke **startu konkrétní autorizační metody** z vybraného scénáře.

Vstupem je JSON objekt obsahující požadovaný typ autorizační metody - **CODE** a všechny elementy specifické pro tento krok.

Výstupem tohoto resource je přehled hodnot potřebných pro dokončení autorizace.

Např. pro CODE odpovídající federované autorizaci bude odpovědí URL a parametry pro přesměrování na federovanou autorizační stránku.

A např. pro CODE odpovídající autorizaci prostřednictvím OTP kódu zaslání SMS bude odpovědí jen potvrzení o odeslání kódu. Samotné odeslání iniciuje banka.

Charakteristika resource

URI:	/payments/{paymentId}/sign/{signId}
HTTP Metoda:	POST
Request URL:	https://api.koba.sk/sandbox/pisp/v1/payments/{paymentId}/sign/{signId}
Authorization:	request vyžaduje autorizaci uživatele/klienta jako součást volání API
Certification:	request vyžaduje použití kvalifikovaného certifikátu třetí strany v rámci navázání oboustranné TLS komunikace se serverem. Třetí strana je identifikována ověřením platnosti a obsahu tohoto certifikátu

Stránkování: ne

Třídění: ne

Filtrování: ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Obsah POST requestu a response volání viz kapitola 0 ELEMENTY ZPRÁVY Krok II. Inicie autorizace platby – specifické pro každou banku

Chybové kódy definované ČBA standardem pro službu POST iniciace autorizace platby:

HTTP STATUS KÓD	ERROR KÓD	ÚČEL
401	UNAUTHORISED	Nevalidní/chybějící access token = uživatel není autentizován
403	FORBIDDEN	Nevalidní/chybějící certifikát = provider není autentizován
404	ID_NOT_FOUND	Požadované id neexistuje

400	AUTH_LIMIT_EXCEED	Tento resource není možné autorizovat touto metodou
-----	-------------------	---

Elementy zprávy Krok II. Inicicace autorizace platby - specifické pro každou banku

Parametry requestu:

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZENTACE
+	authorizationType	[1..1]	PISP ALL	Text	Kód požadované autorizace (z autorizačních scénářů)

Parametry response:

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP PLATBY	TYP FORMÁTU	PREZENTACE
+	authorizationType	[1..1]	PISP ALL	±	Kód požadované autorizace (z autorizačních scénářů)
+	href	[0..1]	PISP ALL	±	Reference pro vyvolání federované autorizace
++	url	[1..1]	PISP ALL	Text	URL link nebo package federované autorizace
++	id	[0..1]	PISP ALL	Text	Případné id pro vyvolání federované autorizace
+	method	[0..1]	PISP ALL	Text	Metoda pro použití href linku u federované autorizace.
+	formData	[0..1]	PISP ALL	±	Volitelný element. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje data pro odeslání v přesměrování na federovanou autorizaci.
++	SAMLRequest	[0..1]	PISP ALL	Text	Volitelný parametr. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje data SAML requestu.
++	relayState	[0..1]	PISP ALL	Text	Volitelný parametr. V případě metody POST federované autorizace (authorizationType=USERAGENT_REDIRECT) element obsahuje relayState pro návratovou hodnotu.
+	signInfo	[1..1]	PISP ALL	±	Informace o autorizaci instrukce
++	state	[1..1]	PISP ALL	Text	Status autorizace transakce v bankou podporovaném formátu
++	signId	[1..1]	PISP ALL	Text	Jedinečný identifikátor aktuální autorizace transakce

Dotaz na dostatek prostředků (POST /accounts/balanceCheck)

Jedná se o zdroj pro zaslání dotazu na dostatek prostředků na konkrétním platebním účtu plátce účtu. Tento resource je autorizován. Souhlas s přístupem k informacím musí být udělen klientem mimo interakci tohoto API ještě před tím, než dojde k použití zdroje.

Charakteristika resource

URI:	/my/payments/balanceCheck
HTTP Metoda:	POST
Request URL:	https://api.koba.sk/serverapi/pisp/v1/payments/balanceCheck
Authorization:	request vyžaduje autorizaci uživatele/klienta jako součást volání API
Certification:	request vyžaduje použití kvalifikovaného certifikátu třetí strany
Stránkování:	ne
Třídění:	ne
Filtrování:	ne

Query parametry requestu: **nedefinovány**

Parametry hlavičky requestu:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .
API-key	Text	Ne	Volitelný řetězec vydaný komunikující třetí straně jako identifikátor volání této strany primárně sloužící jako konfigurační prvek komunikace.
Authorization	Text	Ano	Parametr slouží pro předání access tokenu autentizovaného uživatele spolu s jeho typem.
TPP-Name	Text	Ano	Název původního TPP, které request vytvořilo. Např.: „Star Corporation, a.s.“. V tomto poli jsou podporovány pouze znaky bez diakritiky.
TPP-Identification	Text	Ne	Identifikace původního TPP, které request vytvořilo. Např.: „CZ013574-15“

Parametry hlavičky response:

PARAMETR	TYP	POVINNÝ	ÚČEL
Content-Type	Text	Ano	Specifikace požadovaného formátu přenosu. Z předpokladu technické specifikace tohoto standardu API je v tomto případě primárně podporován formát application/json .

Obsah POST requestu a response volání viz kapitola 7.1. Elementy zprávy Dotaz na dostatek prostředků.

Chybové kódy definované pro službu POST Dotaz na dostatek prostředků

HTTP STATUS KÓD	ERROR KÓD	ÚČEL
401	UNAUTHORISED	Chybějící certifikát.
403	FORBIDDEN	Volání metody, která neodpovídá licenci, nebo neplatný certifikát.
400	FIELD_MISSING	Chybějící povinné pole v requestu.
400	FIELD_INVALID	Hodnota pole není validní.
400	AC02	[InvalidDebtorAccountNumber] – nevalidní identifikátoru účtu v obsahu požadavku.
400	AC09	[InvalidAccountCurrency] – uvedena nevalidní měna požadovaného účtu.
400	AC12	[InvalidAccountType] - typ účtu neodpovídá povoleným typům účtů (např. se jedná o neplatební účet).
403	AG01	[TransactionForbidden] – neexistující souhlas s přístupem k informacím o dostatku prostředků na účtu.
400	AM11	[InvalidTransactionCurrency] – v požadavku je uvedena neobchodovaná/nepodporovaná měna.
400	AM12	[InvalidAmount] – chybně zadaná částka. Např. příliš nízká, nebo vysoká částka, či špatný formát čísla včetně počtu desetinných míst dle ISO 4217.
400	FF01	[Invalid File Format] – nevalidní JSON formát, či jiný

		technický problém se zpracováním dotazu.
400, 50x	NARR	Narrative – obecný důvod pro odmítnutí platby, s doplněním informace o chybě.
400	RF01	[NotUniqueTransactionReference] – neunikátní identifikátor žádosti.
400	RR10	[InvalidCharacterSet] – nevalidní znaková sada v požadavku.

1.1. Elementy zprávy Dotaz na dostatek prostředků

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	exchangeIdentification	[1..1]	Max18Text	Jednoznačná identifikace dotazu
+	card	[0..1]	±	Transakční karta
++	cardholderName	[0..1]	Max45Text	Jméno držitele karty
++	maskedPan	[1..1]	Max30Text	Maskované číslo karty
+	debtorAccount	[1..1]	±	Účet plátce
++	identification	[1..1]	±	Identifikace účtu plátce
+++	iban	[1..1]	IBAN2007Identifier	IBAN
++	currency	[0..1]	CurrencyCode, ISO 4217	Měna účtu plátce
+	authenticationMethod	[0..1]	CodeSet	Metoda ověření klienta
+	merchant	[0..1]	±	Obchodník provádějící transakci
++	identification	[1..1]	Max35Text	Identifikace obchodníka
++	type	[0..1]	Code	Typ obchodníka
++	shortName	[1..1]	Max35Text	Název obchodníka
++	commonName	[1..1]	Max70Text	Jméno obchodníka tak, jak je uvedeno na potvrzení o platbě
++	address	[0..1]	Max140Text	Adresa obchodníka
++	countryCode	[0..1]	CountryCode, ISO 3166 (2 znaková verze kódu)	Země obchodníka
++	merchantCategoryCode	[1..1]	Min3Max4Text, ISO 18245	Kód obchodníka v návaznosti na typ obchodu
+	transactionDetails	[1..1]	±	detaily transakce
++	currency	[1..1]	CurrencyCode, ISO 4217	Měna dotazu na zůstatek
++	totalAmount	[1..1]	Amount	Částka dotazu na zůstatek

1.2. Elementy response Dotaz na dostatek prostředků

ÚROVEŇ	PRVEK ZPRÁVY	VÝSKYT	TYP FORMÁTU	PREZENTACE
+	responselIdentification	[1..1]	Celé číslo	Jednoznačná identifikace odpovědi na dotaz na dostatek prostředků (ze strany ASPSP).
+	exchangeIdentification	[1..1]	Max18Text	Zopakovaná identifikace platební transakce (dotazu na dostatek prostředků) ze strany vydavatele karty, ke které se dotaz na dostatek prostředků na účtu váže.
+	response	[1..1]	Code set	Výsledek dotazu na dostatek prostředků.

1.3. Návratové kódy parametru „response“ – Code set:

KÓD	POPIS
APPR	Dostatek prostředků na účtu
DECL	Nedostatek prostředků na účtu

10. PSD2 slovník - výběr

API – Application Programming Interface

REST - (Representational State Transfer) - REST je architektura API, která nám umožňuje přistupovat k datům a provádět nad nimi CRUD operace. Architektura, která pracuje obvykle přes protokol HTTP/HTTPS. REST je bezstavový, čímž jednak značně zjednodušuje komunikaci s API a umožňuje paralelní zpracování obsahu. Zároveň ho lze dost snadno použít s HTTP, což je velice rozšířený protokol. V neposlední řadě nám poskytuje určitý standard, takže není problém použít cizí API nebo naopak nabízet vlastní API velkému množství dalších uživatelů. Rozhraní REST je použitelné pro jednotný a snadný přístup ke zdrojům (resources). Zdrojem mohou být data, stejně jako stavy aplikace (pokud je lze popsat konkrétními daty). Všechny zdroje mají vlastní identifikátor URI a REST definuje čtyři základní metody pro přístup k nim (GET, PUT, POST, DELETE). Významy jednotlivých HTTP Verbs jsou následující:

- GET - získání dat
- POST - vytvoření
- PUT - úpravy (upraví celý zdroj - chová se jako SET)
- DELETE - smazání
- PATCH - částečné úpravy

REST API - rozhraní pro distribuované prostředí orientované na data, nikoli na volání procedur jako např. RPC (XML-RPC) či SOAP. Webové služby definují vzdálené procedury a protokol pro jejich volání, REST určuje, jak se přistupuje k datům. REST API využívá metody HTTP jako jsou @GET, @PUT, @POST, @DELETE, @PATCH.

TPP – Third Party Provider (třetí strana registrovaná u Slovenské národní banky nebo u jiné oficiální autority v rámci zemí EU)

AIS - Account Information Service

AISP - Account Information Service Provider

PIS - Payment Initiation Service

PISP - Payment Initiation Service Provider

CIS - Card-based Payment Instrument Issuance Service

CISP - Card-based Payment Instrument Issuance Service Provider

ASPSP – Account Servicing Payment Service Provider (banka vedoucí platební účet plátce)